1991-09

Implementation of a configuration management system for a local area network

Brewer, Mack L.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

IMPLEMENTATION OF A CONFIGURATION
MANAGEMENT SYSTEM FOR A
LOCAL AREA NETWORK

by

Mack L. Brewer

September 1991

Thesis Advisor: Norman F. Schneidewind

Approved for public release; distribution is unlimited

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL *(If applicable)* 37	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000	7b ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000
8a NAME OF FUNDING/SPONSORING ORGANIZATION 8b OFFICE SYMBOL *(If applicable)*	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS *(City, State, and ZIP Code)*	10 SOURCE OF FUNDING NUMBERS Program Element No / Project No / Task No / Work Unit Accession Number

11 TITLE *(Include Security Classification)*
IMPLEMENTATION OF A CONFIGURATION MANAGEMENT SYSTEM FOR A LOCAL AREA NETWORK

12 PERSONAL AUTHOR(S) Brewer, Mack L.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT *(year, month, day)*	15 PAGE COUNT
Master's Thesis	From To	September 1991	57

16 SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 COSATI CODES

FIELD	GROUP	SUBGROUP

18 SUBJECT TERMS *(continue on reverse if necessary and identify by block number)*
LAN, Database, LAN Management, Configuration Management

19 ABSTRACT *(continue on reverse if necessary and identify by block number)*
A major concern of system managers on Local Area Network (LAN) environments is to keep track of each of the components and location of network nodes as well as the maintennance history of LAN nodes and accessories. The complexity of the technology and the variety of products used interchangeably make this task particularly hard. This thesis designs and implements a database application to facilitate this effort. It allows the LAN maintenance staff to manage the assets more efficiently and effectively. This system can also be adapted and applied to LAN systems throughout the DoD as required.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT [X] UNCLASSIFIED/UNLIMITED [ ] SAME AS REPORT [ ] DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL Norman F. Schneidewind	22b TELEPHONE *(Include Area code)* 408 646-2494 22c OFFICE SYMBOL As/Ss

Approved for public release; distribution is unlimited.

Implementation of a Configuration Management System for a Local Area Network

by

Mack L. Brewer
Captain, United States Marine Corps
B.S., Texas A&M University, 1980

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
September 1991

ABSTRACT

A major concern of system managers in Local Area Network (LAN) environments is to keep track of each of the components and location of network nodes as well as the maintenance history of LAN nodes and accessories. The complexity of the technology and the variety of products used interchangeably make this task particularly hard. This thesis designs and implements a database application to facilitate this effort. It allows the LAN maintenance staff to manage the assets more efficiently and effectively. This system can also be adapted and applied to LAN systems throughout the DoD as required.

TABLE OF CONTENTS

I. INTRODUCTION

A. LOCAL AREA NETWORK OVERVIEW

Recent advances in computer technology have resulted in a proliferation of automated tasks throughout the business community. One advance that is becoming prevalent in many organizations is the local area network (LAN). Available through many commercial vendors, the LAN links multiple work stations so they can share data and resources throughout the organization. It has created new ways to think of personal computers (PCs) as they have taken on the roles of both server and user. A user is the individual computer that can act as a stand-alone processor or access information or applications from the central "server." Servers hold or control the various network assets which they share with accessing user computers. The shared assets include printers, large hard drives and software, among other things.

Use of a LAN has a variety of advantages for an organization. Expensive resources such as laser printers can be shared by a number of users, reducing redundancy and resulting in sometimes dramatic financial savings. Greater efficiency can be achieved by making information more immediately available on a broad scale. Everyone along the "chain" who wants to review or change a document can do it

electronically, on the same physical file, while sitting at their own work station. Another of the more significant features of the LAN is electronic mail (E-mail) which allows users to manage their own affairs and their subordinates more effectively by sending memos from one work station to another.

This LAN configuration scheme has solved a wealth of communication problems. However, like any new technology, it has also raised new issues that must be dealt with. Someone must track the total system, including such matters as compatibility between machines and accessories, machines and operating systems, machines and application software, and application software and operating systems. Of importance in this paper is the complex problem of maintaining such a system so that it can evolve as new hardware and software become available. A dynamic evolution allows the LAN to continue to enhance the efficiency of the organization rather than cause back-ups and slow down the work flow.

B. NEED FOR A LAN MAINTENANCE SYSTEM

The need to keep a LAN system functioning at high efficiency is obvious. However, the complexity of the technology and the variety of products used interchangeably introduce unique issues. LAN maintenance is a problem not just because of the routine difficulty of maintaining an inventory of items possessed, but also because each machine and piece of software has its own characteristics which should

be documented with an appropriate history. This documentation provides a double value. First, the maintenance person has an accessible history of all parts and changes made to a machine which he can review when the machine suddenly stops working in a given functional area; and second, it provides a reference source for solving a similar problem on a different machine.

By documenting what is learned WHEN it is learned, the maintainer can build a meaningful and useful database. There is a pressing need for the appropriate means by which to document the myriad permutations and combinations of hardware and software. This systems takes a step in providing that means.

1. Hardware Peculiarities

a. Incompatible Compatibles

There is a wide variety of microcomputers available in the marketplace, and any organization that purchases over a number of years from different vendors (such as the federal government) most likely has many different machines. These "clones" represent themselves to be IBM compatible, but are not truly identical in what they do or how they do it. For example, different manufacturers use different basic input-output system (BIOS) chips, and these BIOS chips come in different versions. It makes the blending of various technologies a tremendous challenge.

b. Switch Settings

Many peripheral devices (modems, emulator boards, etc.) are added by installing add-on boards in the node. This is a common-place activity in the PC world, and might be thought of as a simple process in terms of LAN maintenance. However, nearly identical peripherals and even motherboards made by different vendors have different jumper and/or dip-switch settings, and in some cases, may not have settings. One add-on device may or may not be compatible with another device already installed. Interrupt levels, which notify the central processing unit of a transmission, may be in conflict with levels already present on the machine. To complicate matters, some software may not be able to address the full range of settings on a given device. It all adds up to the fact that the operation of the same software from one "clone" to the next can be very different. This reinforces the argument for documenting installation and maintenance activities.

c. Drive Types

Hard disks, too, come in a wide variety. Modified Frequency Modulation (MFM), Run Length Limited (RLL), Integrated Drive Electronics (IDE), Enhanced Small Device Interface (ESDI) and Small Computer System Interface (SCSI) are the likely formats that the maintenance person will encounter. Each format requires a different interface with

the motherboard. For example, the IDE or "AT" drives have the controller physically on the device and interface through an interface card or I/O chip. The principle of device independence which allows such a variety of hard disks promotes great flexibility by allowing users to select from different storage devices based on their needs, wants, and budget, but it adds tremendous complexity to the act of maintaining a LAN composed of such a blend.

2. Software Limitations

Software also has its own set of compatibility problems. A given piece of software may not work with a given BIOS chip. A certain program may require a specific generation (or higher) of an operating system, and each node must deal with two operating systems, DOS and the network system software. These peculiarities, if not well documented, may have to be discovered and rediscovered, solved and resolved, by different maintenance personnel through trial and error.

3. Spare-parts Inventories

Any person performing routine maintenance on a number of machines can benefit from an automated listing of spare-parts holdings. This allows for effective inventory management for high turn-over items and also for the ability to search that inventory for a given item or group of items.

That search, called a query in the database management world, is one of the principal strengths of an automated system.

C. FOCUS OF THIS WORK

The heart of this work is to resolve the problems associated with the complexities enumerated above, and to provide an information base from which better LAN management decisions can be made. The implementation, uses a commercially available database management produw and will assist lab maintenance and management personnel in tracking both maintenance actions and hardware and software configurations on a LAN. It focuses on IBM-compatible, DOS-based machines common throughout the Department of Defense.

The system was developed and implemented for the Administrative Sciences (AS) Department at the Naval Postgraduate School as a first iteration of the configuration management program. Heretofore, LAN maintenance in the AS Department was done semi-automatically, and was a cumbersome process for LAN management and maintenance personnel. The automated system was implemented to maintain one of the school's IBM Token-ring Networks, which is one of the more successful commercially-available LANs.

A previous thesis was completed at the Naval Postgraduate School in September of 1989 by Douglas A. Suriano which identified systems requirements for the AS Department and provided an application design for a LAN management system

(Suriano, 1989). Using Suriano's work as a base, with some needed modifications, the maintenance system was developed for immediate use within the AS Department but with the capability of being used throughout the DoD as needed.

The system was coded using software currently available and working on the Token-ring and widely available in DoD. The appendices to this thesis and the commented code will together provide high-quality documentation of the system to allow continued study and improvement of the system.

D. ORGANIZATION OF STUDY

This study is organized as follows. Chapter II overviews the previous design, while Chapter III provides the modifications to Suriano's design and the justification for those modifications. Chapter IV describes the implementation process. Conclusions and recommendations are offered in Chapter V.

II. OVERVIEW OF PREVIOUS WORK

A. INITIAL REQUIREMENT FOR SYSTEM

Because of a need within the Administrative Sciences Department (AS) to maintain various LANs for education and research, the department wanted a system to automate the configuration management of hardware and software. To this end Suriano worked with the LAN maintenance personnel to determine system functional and data requirements (Suriano, 1989). He developed an initial design of data structures and relationships and an application design. Suriano followed an object-oriented approach and followed the traditional pattern of a Definition Phase, a Requirements Phase, and a Design Phase. This work constitutes the Implementation Phase, and the Maintenance Phase should be a topic for follow-on study.

B. OBJECT-ORIENTED APPROACH

In the object-oriented approach, users define their requirements in terms of the physical entities around them. For the LAN administrator, these include workstations (nodes); disk drives, printers, etc. (accessories); and software (both system and application). By relating data to the physical entities around them, users can better assist in defining the data elements that must be included in an application designed for their work area.

Once the objects are identified, the properties pertinent to these objects can be defined. For example, if the object were a node, the characteristics might include a serial number, the central processing unit (CPU) speed, and the video display type. The objects, along with their properties and the relationships among the objects which have been identified, are transferred into relations, attributes and relationships among relations. The user Requirements Phase also requires the creation of dataflow diagrams.

C. DESIGN PHASE

In the logical database Design Phase, the objects developed in the requirements phase are transformed into relations. Each relation could be thought of as a table with each row representing a node and with columns across that row containing the node's "values" for each attribute, i.e. 225564, 33 MHz, VGA. If a new node were added to this database, a new row (or tuple) would be added to the table.

If there are a number of objects in the application environment, such as LANs, accessories, software, etc., there would be one or more relations or tables for each object. Tables should be designed using normalization techniques to make sure that the data is not corrupted by adding, deleting, or modifying a row (or record). This also aids in the identification of objects less obvious to the user. The entire collection of these normalized tables or "relations"

constitutes the database. (Kroenke, 1988) Table I shows Suriano's initial relation definitions.

RELATION	ATTRIBUTE	LEN.	TYPE
LAN	LAN-ID	2	N
NODE	Node-Name	6	A
	CPU-Model-#	10	A/N
	CPU-Serial-#	10	A/N
	Display-Model-#	10	A/N
	Display-Adapter-Type	3	A
	Keyboard-Type	10	A
	Motherboard-Memory	4	A/N
	Expanded-Memory	4	A/N
	Extended-Memory	4	A/N
	Network-Board	10	A
	Hard-Disk-1	4	A/N
	Hard-Disk-2	4	A/N
	Floppy-Disk-Drive-1	12	A/N
	Floppy-Disk-Drive-2	12	A/N
ACCESSORY	Accessory-Name	10	A
	Accessory-Type	10	A/N
	I/O-Port	10	A/N
SETTING	Switch-Setting	8	N
	Comment	200	A/N
APPLICATION-SOFTWARE	Application-Name	15	A/N
	Version	4	A/N
SYSTEM-SOFTWARE	System-Software-Name	15	A/N
	Version	4	A/N
CABLE-PLANT	Item-Name	20	A/N
	Item-Quantity	2	N
SPARE-PARTS	Spare-Name	15	A/N
	Spare-Quantity	2	N
	Location	10	A/N

Table I

To complete the Design Phase, the data flows are transferred into a menu hierarchy (Figure 1 contains the menu hierarchy from the initial design) which provides the needed actions on the data and data materializations to display the

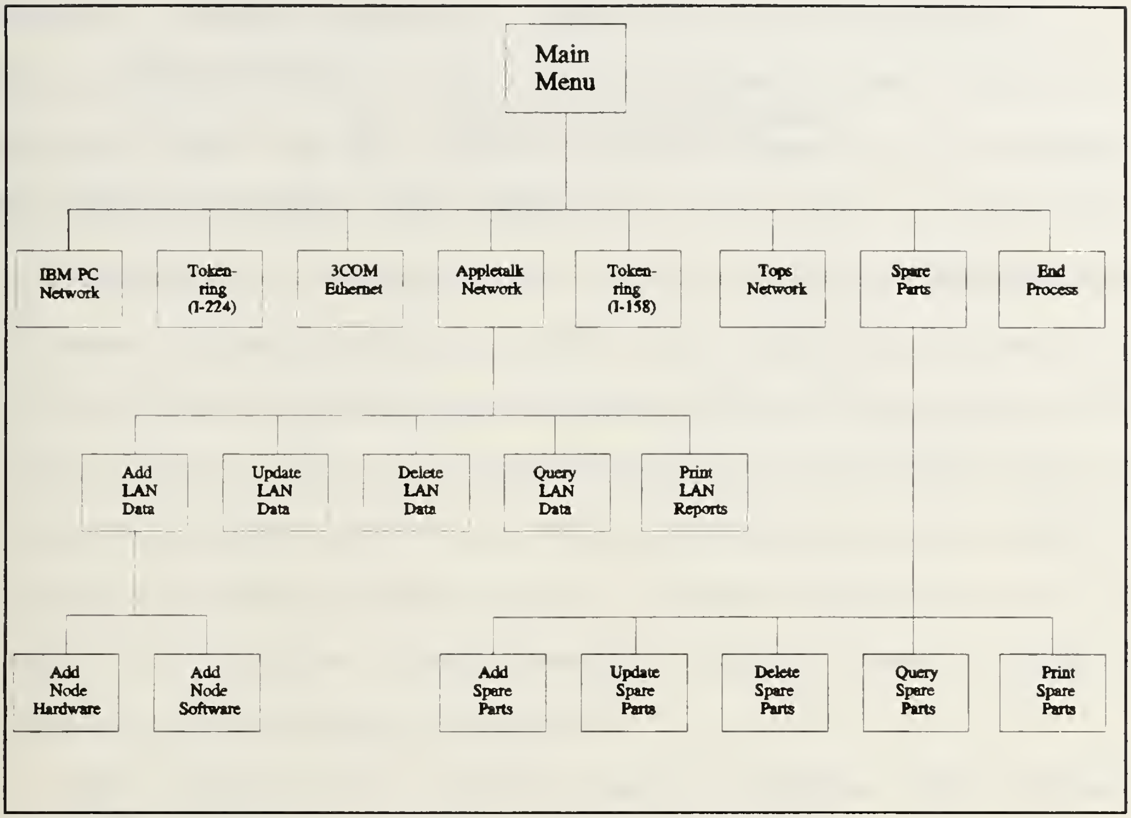


Figure 1 - Initial Design of Menu Hierarchy

data. The menu selections and data materializations, along with access to tools provided by the database management system (DBMS) allow manipulation of the tables. Selections would include adding a row (or record), deleting a record, editing a record, and querying the database for a record or set of records. A query might show a list of nodes on a certain LAN, or nodes with a certain minimum CPU speed.

While this is a simplified explanation, it portrays the general methodology Suriano followed in defining objects and tables in the Requirements Phase of the LAN application environment.

Six networks were included in Suriano's model, including an IBM PC, two (location) Token-rings, a 3COM Ethernet, a Tops network (for communicating between IBM and Apple) and the Appletalk. Only one of the Token-ring networks (I-224) was implemented.

III. DESIGN MODIFICATIONS

A. NEW FUNCTIONAL REQUIREMENTS

During the implementation of the system, the need for changes became evident. There was disparity between the initial design and the prioritized desires of the user as set forth below, with the ultimate goal of the system being to provide a sound base for a decision support system (DSS) to support the maintenance effort. Items listed as "will" or "must" are system requirements. Those items listed as "should" are desirable features.

- The system will provide both hardware and software assistance.
- It must be able to display the information needed for a node.
- It must incorporate the pertinent attributes of the LAN.
- Ultimately it should provide an expert system to provide prompts to isolate problems and to assist in actions such as LAN set up, node addition, adding a board, and maintenance routines.
- It should have a plain language interface.
- It should be able to present the network in a hierarchical fashion.
- It should be adaptable to different networks.

(Schneidewind, 1991)

B. REQUIREMENT FOR GREATER HARDWARE DETAIL

New details became germane to the proper implementation of the system, since the Suriano design (probably developed without benefit of this DSS goal) would not provide sufficient tabular information. The random access memory (RAM) on a given node, DOS version, the BIOS manufacturer, and the hard drive type are examples of the data elements that would have to be included in a meaningful DSS.

1. Node Attributes

The attributes (DOS version, etc.) listed above are among the object characteristics that were missing from or inadequately characterized in the initial design. There was also a need to change the length and character of the data type of certain attributes. These changes will make the system more robust in dealing with different generations of equipment.

2. Secondary Storage Devices as Accessories

The old design calls for drives to be attributes of the node. Due to the varying number and type of hard and floppy drives and other storage devices available for machines, they should be handled as accessories. While a machine will almost always have certain single elements such as the keyboard and monitor, there is no guarantee that there will be a set number or particular blend of storage devices. Machines may or may not have hard drives, and they may have

two rather than one. Other machines might have one or two floppy disk drives, and might have a CD ROM and/or tape backup. The possible mixes are endless, and the system is strengthened by treating such devices as accessories, which allows unlimited combinations.

3. Drive Type Parameters

No better example exists for the requirement of greater hardware detail than the wide variety of information required on hard disk drives. The number of heads, cylinders, and sectors varies from drive to drive. When a machine is "set up," the user must specify this information and input it into the CMOS (older machines did not allow this set up as they held no battery-powered memory). Specification of drive parameters can be made using a predefined drive type, or where the setup program allows it, by specifying the information under a "user defined" number.

There are a number of predefined drive types, but there is a lack of standardization within the industry, which increases the need for ample documentation. Tables II and III display partial drive type and parameter listings for an AT machine using a Phoenix BIOS, and the corresponding drive types as listed in QAPlus, a commercial quality assurance tool. The tables are identical through drive type 23, where suddenly the differences are substantial. The future quest for larger, quicker drives, sold by more and more vendors will

Type	Cyls	Heads	Secs	Write Precomp	Landing Zone
1	306	4	17	128	305
2	615	4	17	300	615
3	615	6	17	300	615
4	940	8	17	512	940
5	940	6	17	512	940
6	615	4	17	0	615
7	462	8	17	256	511
8	733	5	17	0	733
9	900	15	17	0	901
10	820	3	17	0	820
11	855	5	17	0	855
12	855	7	17	0	855
13	306	8	17	128	319
14	733	7	17	0	733
15	-- Reserved --				
16	612	4	17	0	663
17	977	5	17	300	977
18	977	7	17	0	977
19	1024	7	17	512	1023
20	733	5	17	300	732
21	733	7	17	300	732
22	733	5	17	300	733
23	306	4	17	0	336
24	925	7	17	0	925
25	925	9	17	0	925
26	754	7	17	754	754
27	754	11	17	0	754

Table II - Phoenix AT BIOS Drive Type Table

make standardization more difficult, and the recorded descriptive information will be even more relevant for the drive installer.

C. NEW DESIGN

1. Object Diagrams

Appendix A includes revised system object diagrams. Properties that were not previously identified have been

Type	Cyls	Heads	Secs	Precomp	Landing
1	306	4	17	128	305
2	615	4	17	300	615
3	615	6	17	300	615
4	940	8	17	512	940
5	940	6	17	512	940
6	615	4	17	0	615
7	462	8	17	256	511
8	733	5	17	0	733
9	900	15	17	0	901
10	820	3	17	0	820
11	855	5	17	0	855
12	855	7	17	0	855
13	306	8	17	128	319
14	733	7	17	0	733
15	--	Reserved	--		
16	612	4	17	0	663
17	977	5	17	300	977
18	977	7	17	0	977
19	1024	7	17	512	1023
20	733	5	17	300	732
21	733	7	17	300	732
22	733	5	17	300	733
23	306	4	17	0	336
24	1024	7	17	-1	1024
25	615	4	17	0	615
26	1024	4	17	-1	1024
27	1024	5	17	-1	1024

Table III - QAPlus Drive Type Table

added, and irrelevant properties dropped. Also refined were the relationships of the objects to other objects. The spare parts object was found to be unneeded since a spare is simply an unassigned instance of an accessory.

2. Relational Diagram

Naturally, the changes in the object diagrams alone required commensurate changes in the relation diagram. The diagrams provide evidence that the node is the focus of the

LAN managers environment. All other relations are related to the node relation. The relation diagram is presented in Appendix B.

3. Relation Definitions

The new relation definitions are placed in Appendix B and reflect the needed changes to make the system more responsive to both the current PC LAN environment and to any future utilization of the same data structures (such as the DSS).

4. Menu Hierarchy

The DBMS selected allows the use of pull-down and pop-up menus. These afford the novice more help through the descriptive nature of the prompts and the intuitive movement through the menu structure (the user can be "down" in the menu chain and still see his options for other actions). The pull-downs and pop-ups designed are presented in Appendix D and will be described in greater detail in Chapter IV, Section E.

5. Data Forms and Materializations

Appendix E includes the format for input and modification of information related to various LANs and their nodes. More particulars will be provided in Chapter IV, Section E.

IV. IMPLEMENTATION

A. METHODOLOGY

1. Prototyping

The system was implemented as modified in Chapter III as a prototype based on the new design, (APPENDICES A through F). It is geared toward the Token-ring environment described below, but because of its modular design, any function can be improved or added and simply replaced on or appended to the system. This will facilitate its adaptability to other LAN configurations.

2. Database Management Concerns

a. Concurrent Processing Control

While multiple-user application systems in a LAN environment have to be provided with record locking to prevent more than one user from accessing the same data records at the same time, no special considerations regarding locking records for this maintenance had to be taken because this is designed as a single-user system. If additional persons begin to use the system, precautions for the DBMS use in a network environment must be heeded.

b. Data Recovery

Like most PC-based systems, Ashton-Tate's dBASE IV version 1.1, the database management system to be used,

provides no protection to the user if large amounts of entered data is contaminated or lost. There is software commercially available to aid in data recovery, but the cost is not likely justified because of the stable nature of the database in this system. Backups to another medium are recommended weekly and whenever an inventory is taken to ensure that valuable trouble-shooting information and hardware data is not lost. A duplicate set of databases will be created on the application disk whenever the system is exited properly.

c. Security

Security of the system is an issue because unwanted access to the system could cause extensive problems. No precautions were warranted for the initial prototype, since it is a single-user system, however like concurrent processing, if multiple LAN maintenance personnel are to have access to the system in a network environment, security is an issue. The addition of the command language interface will complicate security of the program, so it must be controlled by directory access security.

B. DBMS SELECTION

DBASE IV version 1.1 was selected as the implementation database management system (DBMS); it will be referred to as simply dBASE IV in this document. DBASE IV is readily available commercially and was already in place on the Token-ring LAN. Its data structures are commonly usable by other

applications such as DSSs and spreadsheets. It is a relational DBMS and is appropriate for the design techniques developed. DBASE IV is completely compatible with the Token-ring technology, and most popular networks. It can be used in a network mode and has record-locking provisions to prevent concurrent access of records.

C. NORMALIZATION

Normalization is the way in which tables (called relations) are formed in the relational data model. Fully-normalized tables represent the optimal method for data storage (Date, 1986). The accessories table is not fully normalized, in that attributes have been maintained in the table which do not pertain to all accessories. Relation definitions are included in Appendix C. This is an admitted inefficiency for purists, but the monumental leap in coding complexity that would be required to display a node with all of its accessories or to handle queries and reports fully justifies such a deviation.

D. CHARACTERISTICS OF THE TOKEN-RING

This maintenance system is intended to be adaptable to any manufacturer's LAN. Proficiency in the special characteristics of another LAN (differing from the IBM Token-ring used here) would be the responsibility of the LAN maintenance person. By utilizing the accessory capability of

the system, any items required by each node can be listed as belonging to that node.

The IBM Token-ring utilizes multiple access units (MAUs) with eight connectors each, to connect individual user and server nodes. Each node has a Token-Ring adapter network board installed in an expansion slot to which a nine-foot adapter cable is fastened. This cable hooks directly to the MAU, or if required, a patch cable can extend the adapter cable to reach the MAU. The Token-ring is a very versatile arrangement, as nodes can be added to and removed from the network with no interruption in network services.

Both user and server nodes exist in the Token-ring. In this system the user nodes are AT clones (80286 processors running at 10 MHz) and XT clones with accelerator boards containing 80286 processors running at 7.2 MHz. There are two AT-clone server nodes, and to maximize software availability (because of limited disk space) they have different application software on them.

User nodes share the resources of the appropriate server. A third (XT-clone) server acts as a gateway and allows connection to the school's mainframe computer as a 3270 emulator; up to five users can run sessions concurrently. Ten of the user nodes have the software for 3270 emulation. Other user nodes have modems, and one user-instructor computer has both. There is also a two-drive bernoulli box (ten MB each)

on the gateway server to allow handling of large files and programs in this environment.

The AT-clone user nodes have EGA color monitors, the XT-clone user nodes have CGA color monitors, and the server nodes have monochrome monitors. Each server drives an IBM Proprinter. The AT server nodes have 1 megabyte of RAM, the rest have 640 kilobytes of RAM.

E. USER INTERFACE

1. Menu Presentations

Access to functions in the LAN maintenance system are provided to the novice and intermediate user through a light-bar menu system of pull-downs and pop-ups. The menus (Appendix D) follow the object/ action approach, allowing the user first to select the object he wants to deal with (i.e. LAN or node) and then to select the desired action (such as add or modify). This is a logical chaining and keeps the user channeled for the desired action.

2. Menu Actions

The initial presentation is a bar menu with pull-downs attached. These top level choices are LANS, Spares, Maintenance and Quit. Moving the cursor left or right to the desired choice allows selection from the associated subordinate pull-down menu. Selections made on pull-down or pop-up menus call a subroutine or another bank of choices in

a pop-up menu. Items can be selected by highlighting the desired choice with the cursor arrows the pressing the enter key, or whenever first-letter choices are unique, by pressing the first letter (Pressing '3' on Menu 1 would select the 3COM Ethernet for subordinate actions). Detailed actions for each menu are included in Appendix D.

3. Input Screens

Input screens are easily readable, implemented as full-screen fill-in-the-blank forms (Appendix E). Where possible, entry fields are filtered or pick lists are used so that only acceptable inputs can be entered into the database. This helps to maintain data integrity and to enforce domain and intra-relation constraints. Care was taken to place the same data item in the same location on the screen whenever possible to speed the use of the screens, and colors and heightened screen intensity are used on data items. Shown are the node screen (with and without accessories) and an accessory screen. On each screen the appropriate network and node are visible to the user to lessen the chance of action on the wrong node.

F. CODING CHALLENGES

1. Problems with Deleting a LAN or a Node

If the maintenance person removes a node from the LAN, the basic node and its associated accessories should be returned to inventory. This is an automated function but

requires validation from the user to prevent a non-functioning part (perhaps the reason the node was removed from the LAN) to be added the spare parts inventory. The DBMS selected does not enforce these types of inter-relation constraints, so they had to be addressed manually through coding.

Of equivalent complexity is the problem of removing an entire LAN from the system when multiple LANs are in use. Heretofore, the LAN has been addressed as a single entity, but many organizations have multiple LANS which themselves can be linked. This may be less likely to occur in most DoD or business environments than in an academic community, but allowing for this eventuality makes for a robust system. The system allows a LAN to be removed; however, it first requires that all nodes of the LAN are removed from the LAN (or equivalently transfe^Xxed to another LAN) before such deletion is allowed. This protection also had to be explicitly coded.

2. Addition of Other LANS

Adding a LAN is a two-step process as well: the LAN itself must be added, and then one-by-one the nodes must be added. If nodes are transferred from another LAN, their accessories can remain with them. However, the user must validate each accessory to prevent blindly adding data on an invalid network board to the new LAN.

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

Employment of this LAN maintenance system will simplify the life of LAN managers and maintenance personnel by providing the means to record information about the machines, a system through which to query them, and a knowledge base on which to base future decisions. Not all of the design goals have been met. Handling the hierarchical nature of this environment where multiple LANs have a number of nodes each with a number of accessories of many different types made the task quite complex.

DBASE IV version 1.1 has an elaborate application generator to assist in providing a prototype application quickly, but it was very awkward to use to input anything other than the most basic browse and edit functions. One of the most limiting was that even the most minor change to the selections required complete regeneration of the code. Because of its relational nature, it did allow normalization of the data. The limits on the number of files and screens that can be open simultaneously posed barriers to writing modular code.

DBASE IV data structures are widely compatible or importable to other DBMSs, spreadsheets, and decision support

systems. The great amount of detail captured by this system is an asset that should not be overlooked in the general areas of physical security, accountability, purchasing, and resource management. It should not be overlooked as a store of corporate knowledge. Effective use of the system can replace a good deal of paper (which can be out of date as soon as printed) with a dynamic, on-line system, updated as changes are made. It allows maintainers to leave their expertise with the system.

B. RECOMMENDATIONS FOR FURTHER STUDY

1. Using Software to Retrieve Configuration Data

There are a number of commercially-available products used as diagnostic utilities for PCs that are able to scan a PC and determine devices that are present, their type, BIOS versions, drive types and interrupt levels. They come in a great variety with differing levels of detail, but they represent an enhancement to most means of determining a machine's configuration. Use of these tools to capture data for a maintenance system would greatly ease implementation of a program like this one at a new sight by eliminating data entry errors and drudgery.

2. Implementing a Decision Support System

As mentioned before, the high level of detail that can be stored in this system, along with dBASE IV's compatibility with so many other programs (like VP Expert a readily

available DSS), make it an ideal knowledge base for implementing a DSS. By implementing such a DSS, a maintainer can go beyond the simple query capability provided by this program. The system might be built to answer questions posed such as "Can I add this device to this machine?"; "What interrupt level or serial port should I use?"; "What other installed devices might be affected?"; or "Do we have another machine with this identical configuration?"

3. Enhanced System Security

If the system is to be utilized in a network environment, a baton-passing security system with encrypted files should be employed. This would prevent users from running submodules and procedures without going through any security implemented in the opening screens. DBASE IV's descriptive error messages would make it easy for a hacker to identify an alternate way to enter the system, or even more easily to use (or worse, corrupt) the data directly if it is not encrypted.

4. Command Line Interface

As users become more familiar with a software system, the menuing that they once held in high esteem as a novice can become cumbersome, especially on older-generation (slower) machines. An enhancement to the current design would be the addition of a command line interface to allow the expert user to move directly to the module he wants without having to step

through the menus. This enhances "user friendliness" in the software, which will lead to more willing acceptance of the system. The command-line interface could take advantage of the dBASE IV "dot prompt" commands where the user type "do" followed by a space and the name of the module to execute.

APPENDIX A - System Object Diagrams

Node_Name
CPU_Model_#
CPU_Speed
CPU_Serial_#
Motherboard_Memory
Expanded_Memory
Extended_Memory
Cache_Memory
BIOS_Maker
BIOS_Version
Video_Adapter
Monitor_Serial_#
Keyboard_Keys
Keyboard_Compatibility
User_Server

Accessory MV

Application-Software MV

System-Software MV

Cable-Plant MV

LAN

NODE

Accessory_ID
Accessory_Name
Accessory_Type
I/O_Port
Standard_Drive
Drive_Letter
Drive_Type
Heads
Cylinders
Sectors
Landing_Zone
Write_Precompression
Location
Switch_Settings
Comments

Node

ACCESSORY

System_Software_Name
Version
Developer

Node

SYSTEM-SOFTWARE

System Object Diagrams
(cont'd)

Software_Name
Version
Developer

Node

APPLICATION-SOFTWARE

Cable_ID
Item_Name
Item_Quantity

Node

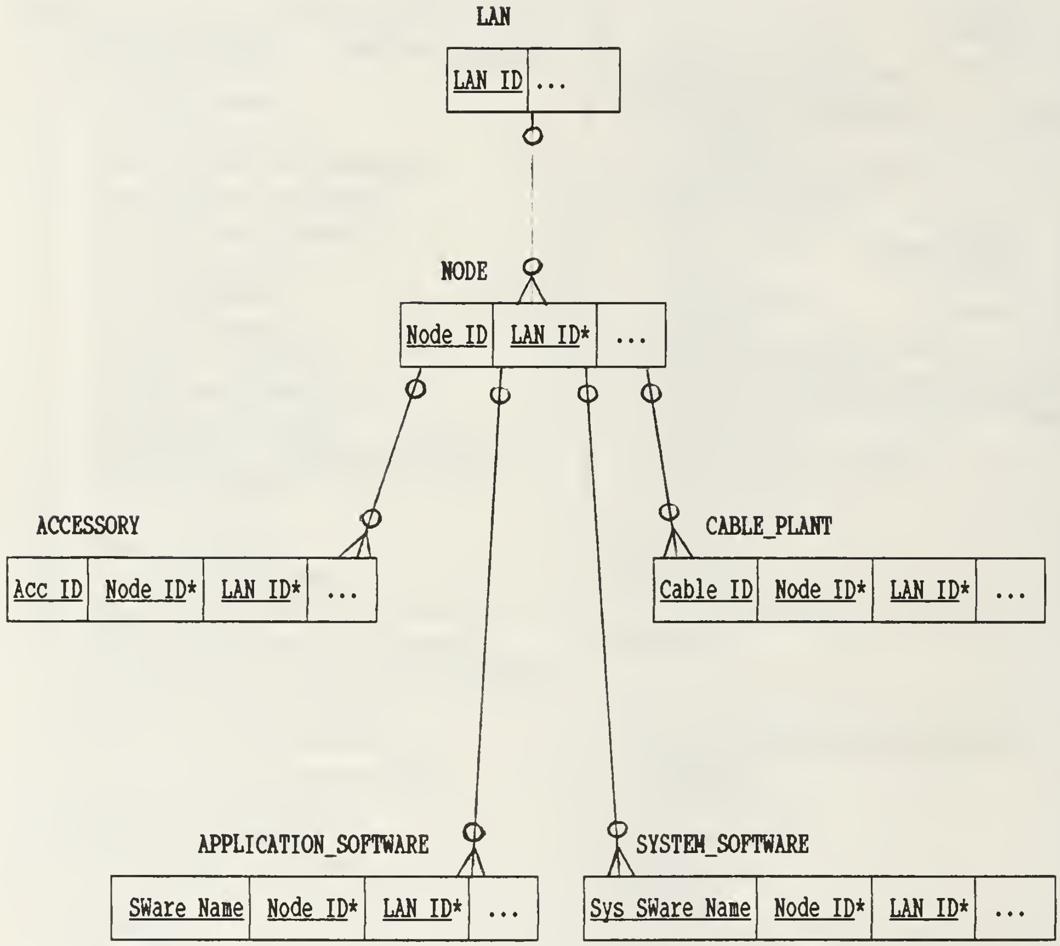
CABLE_PLANT

LAN_ID
LAN_Name

Node
MV

LAN

APPENDIX B - Relationship Diagram



APPENDIX C - Relation Definitions

RELATION	ATTRIBUTE	LEN.	TYPE
LAN	LAN_ID	2	A/N
	LAN_NAME	30	A/N
NODE	LAN_ID	2	A/N
	Node_ID	2	A/N
	Node_Name	6	A/N
	CPU_Model_#	5	A/N
	CPU_Speed	2	A/N
	CPU_Serial_#	10	A/N
	Motherboard_Memory	4	A/N
	Expanded_Memory	4	A/N
	Extended_Memory	4	A/N
	Cache_Memory	3	A/N
	BIOS_Maker	10	A/N
	BIOS_Version	5	N
	Monitor_Serial_#	10	A/N
	Video_Adapter	3	A/N
	Keyboard_Keys	3	A/N
	Keyboard_Compatibility	2	A/N
	User_Server	1	A/N
ACCESSORY	LAN_ID	2	A/N
	Node_ID	2	A/N
	Accessory_ID	2	A/N
	Accessory_Name	12	A/N
	Accessory_Type	10	A/N
	I/O_Port	4	A/N
	Standard_Drive	1	L
	Drive_Letter	1	A/N
	Drive_Type	2	N
	Heads	2	N
	Cylinders	4	N
	Sectors	2	N
	Landing_Zone	4	N
	Write_Precompression	4	N
	Location	10	A/N
	Switch_Settings	8	A/N
	Comments	-	Memo

RELATION	ATTRIBUTE	LEN.	TYPE
APPLICATION-SOFTWARE	Software_Name	15	A/N
	LAN_ID	2	A/N
	Node_ID	2	N
	Version	5	A/N
	Developer	15	A/N
SYSTEM-SOFTWARE	System_Software_Name	15	A/N
	LAN_ID	2	A/N
	Node_ID	2	N
	Version	5	A/N
	Developer	15	A/N
CABLE-PLANT	CABLE_ID	2	A/N
	LAN_ID	2	A/N
	Node_ID	2	A/N
	Item_Name	20	A/N
	Item_Quantity	2	A/N

APPENDIX D - Menu Screens

LANMAINT

Navy Postgraduate School
Administrative Sciences Department
Local Area Network Maintenance Program

Press <┘ to continue.

Welcome Screen

This screen appears briefly upon entry to the application.

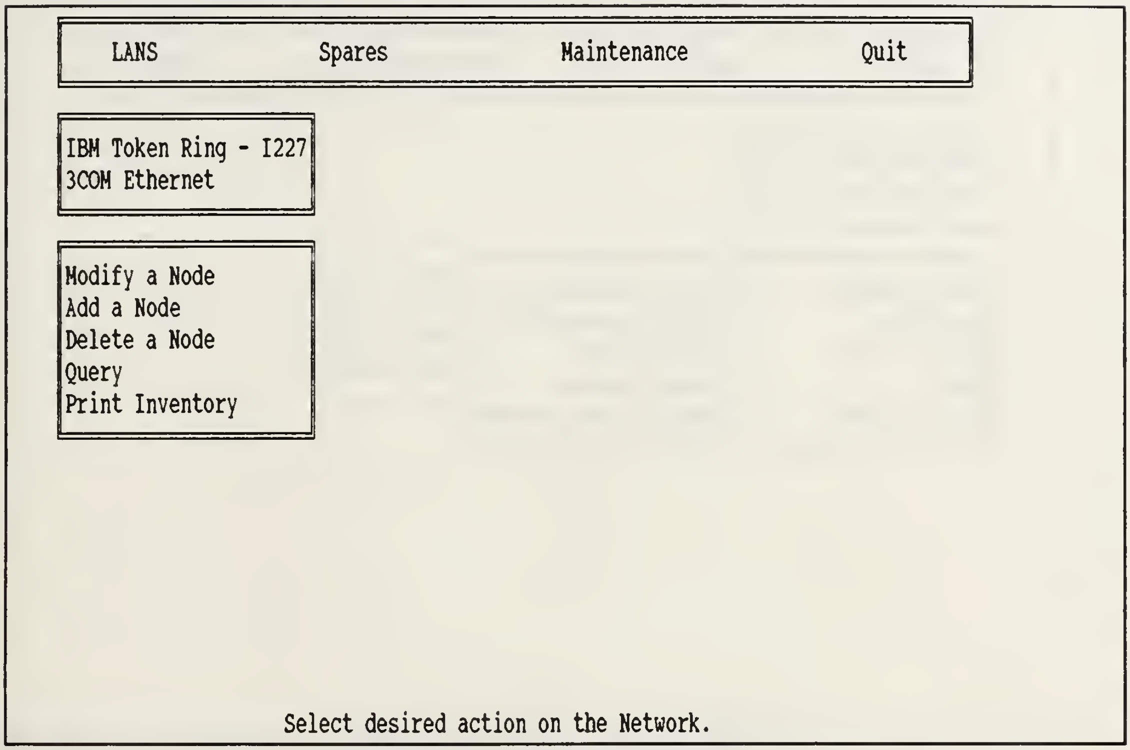
LANS Spares Maintenance Quit

IBM Token Ring - I227
3COM Ethernet

Select the desired LAN.

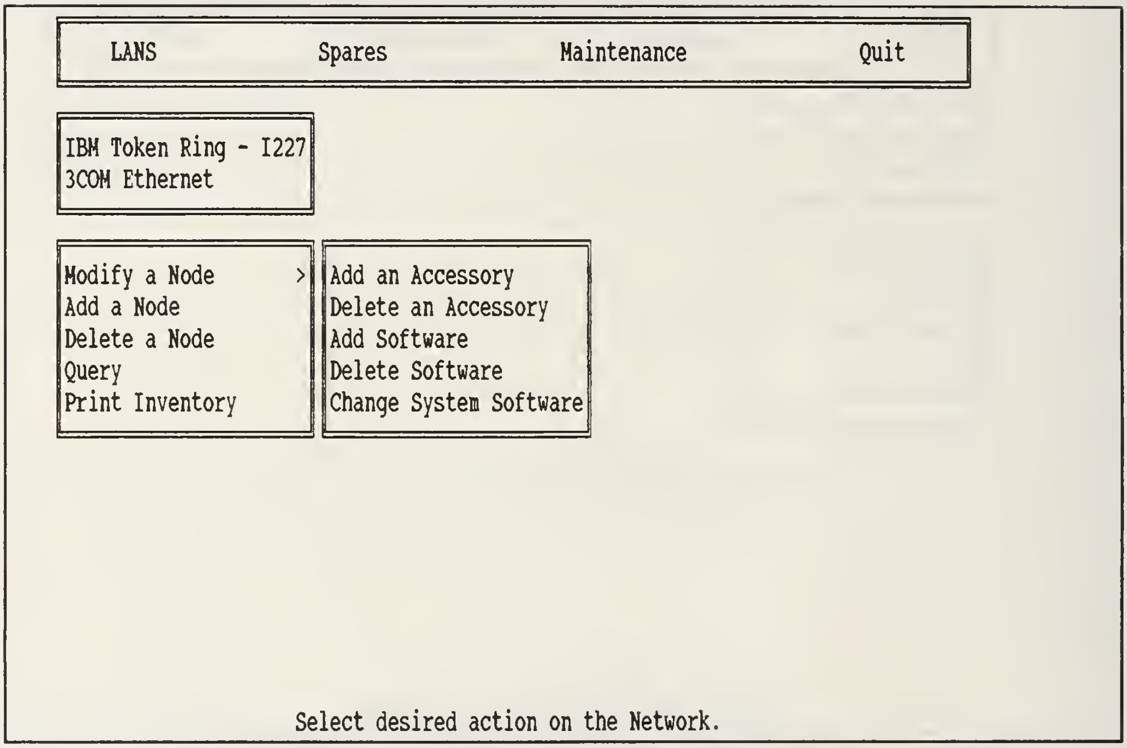
Menu 1 - Main Menu and LAN Pull-down Selection Menu

This menu is the first menu displayed when the user is allowed selections. It will be the most often selected. Any subordinate actions will act upon only the selected network's equipment.



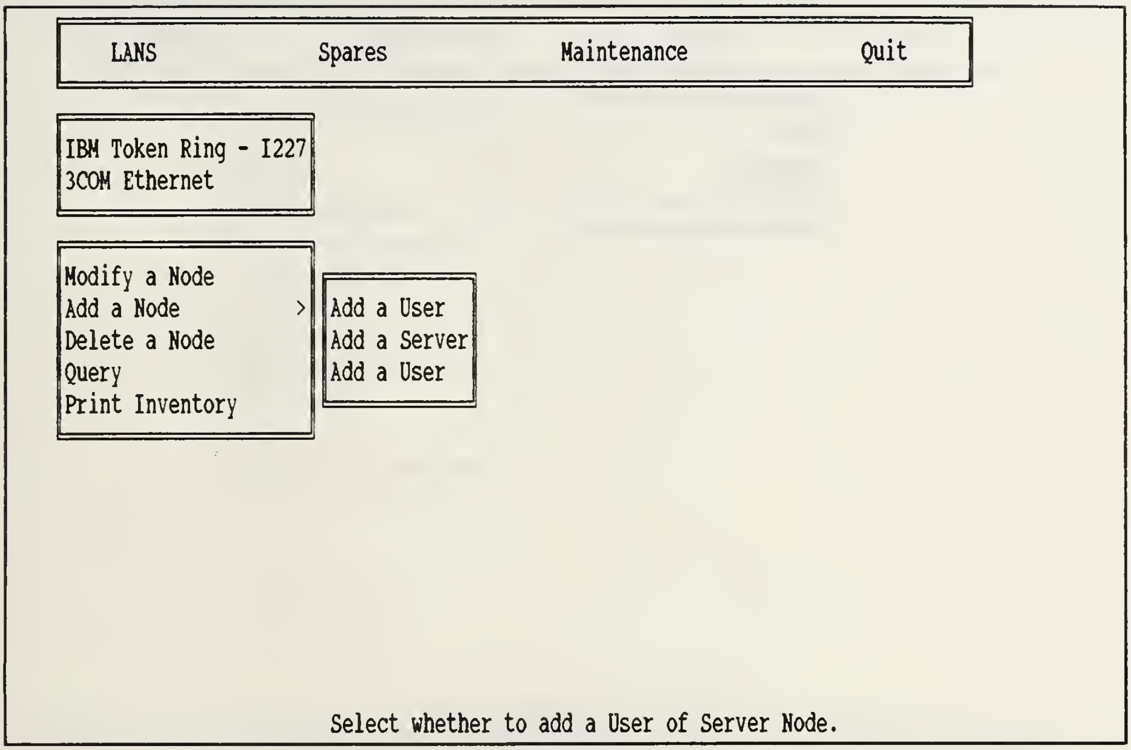
Menu 2 - Select LAN Action Pop-up Submenu

After the LAN is selected, a pop-up with five choices appears. "Modify a Node" and "Add a Node" call pop-up Menu 3. "Delete a Node" runs a validated deletion process. "Query" activates the DBMS query capability. "Print Inventory" prints a node-by-node inventory of the selected LAN.



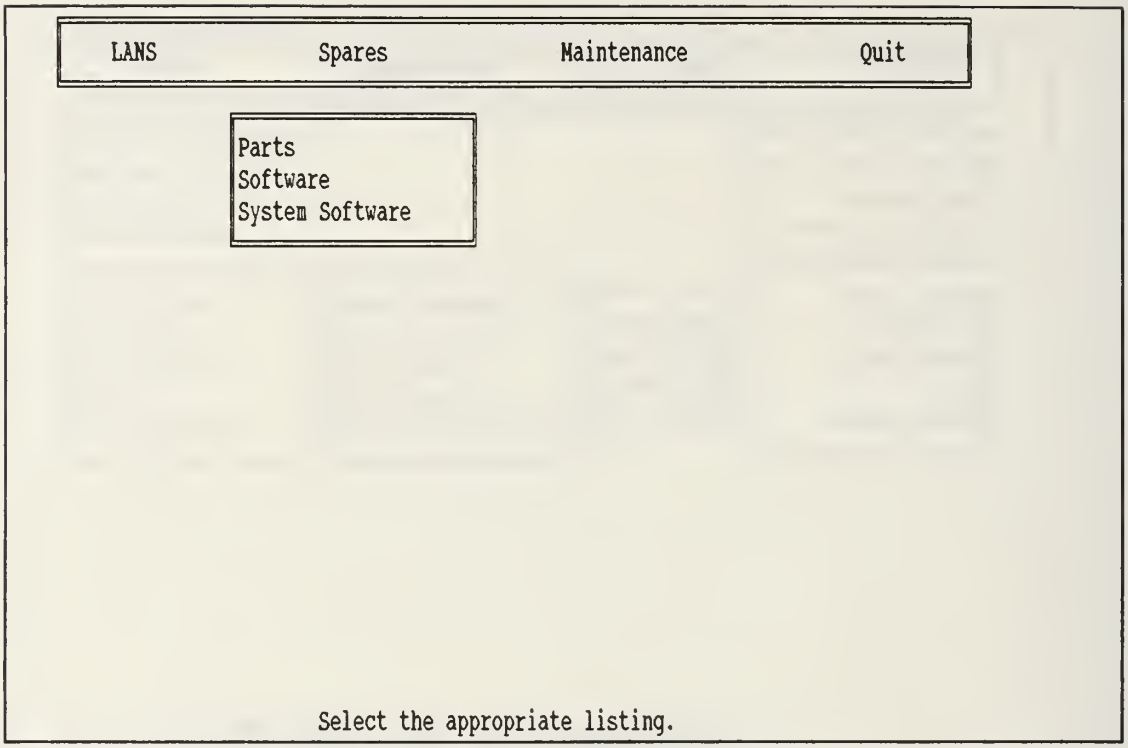
Menu 3 - Modify a Node Pop-up Submenu

This pop-up allows the user to modify a selected node. "Add an Accessory", "Delete an Accessory", "Add Software", "Delete Software," and "Change System Software" all call subroutines and are self descriptive.



Menu 4 - Add a Node Pop-up Menu

This pop-up requires the user to specify the addition of a user or server node. Node numbers are assigned the next sequential number by the program. Either menu choice calls a sub-routine to add a node.



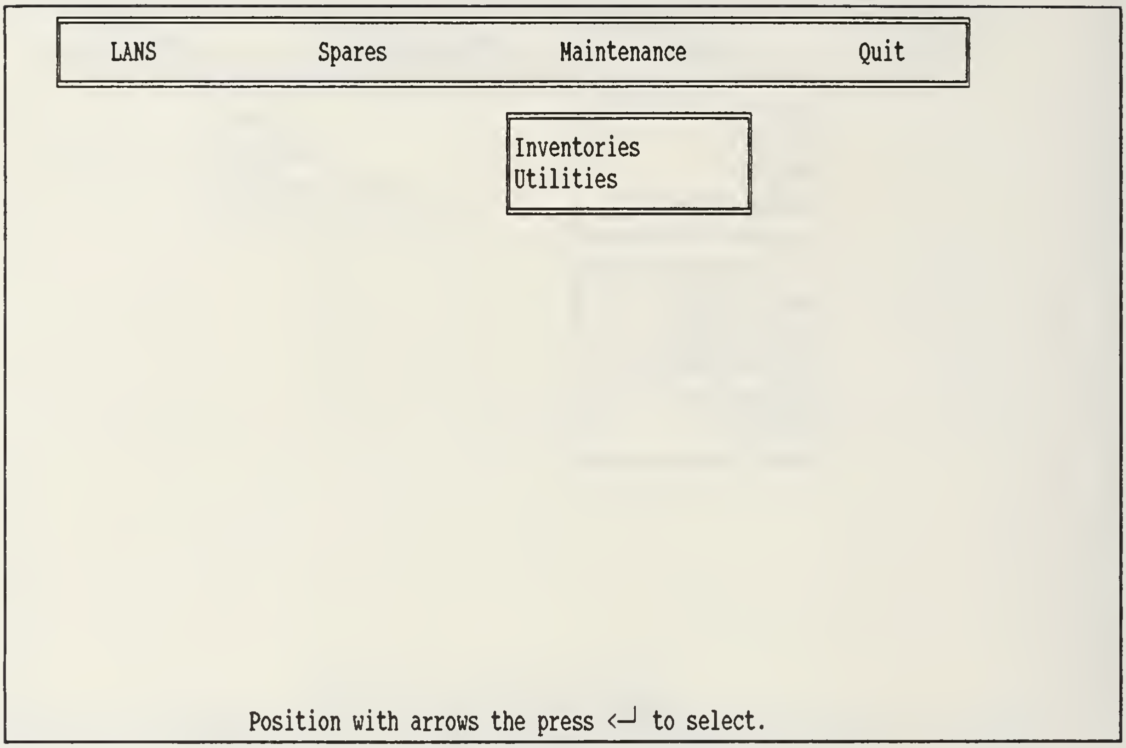
Menu 5 - Spares Pull-down Menu

This bar and pull-down combination displays the selection of hardware and software tracked by the system. The selection of"Parts" (hardware), "Software", or "System Software" all activate Menu 6, however, the actions of that menu are constrained to the selected item.

```
LANS          Spares          Maintenance          Quit

              Parts
              Software
              System Software

              Query Listing
              Add an Item
              Correct Listing
              Use an Item
              Inventory

                  Enter dBASE IV query generator.
```

Menu 6 - Spares Sub-menu Pop-up

All selections on this pop-up call self-described sub-routines except "Query Listing" which grants access to the DBMS query generator. The "Inventory" selection provides a variety of inventories available to the user.



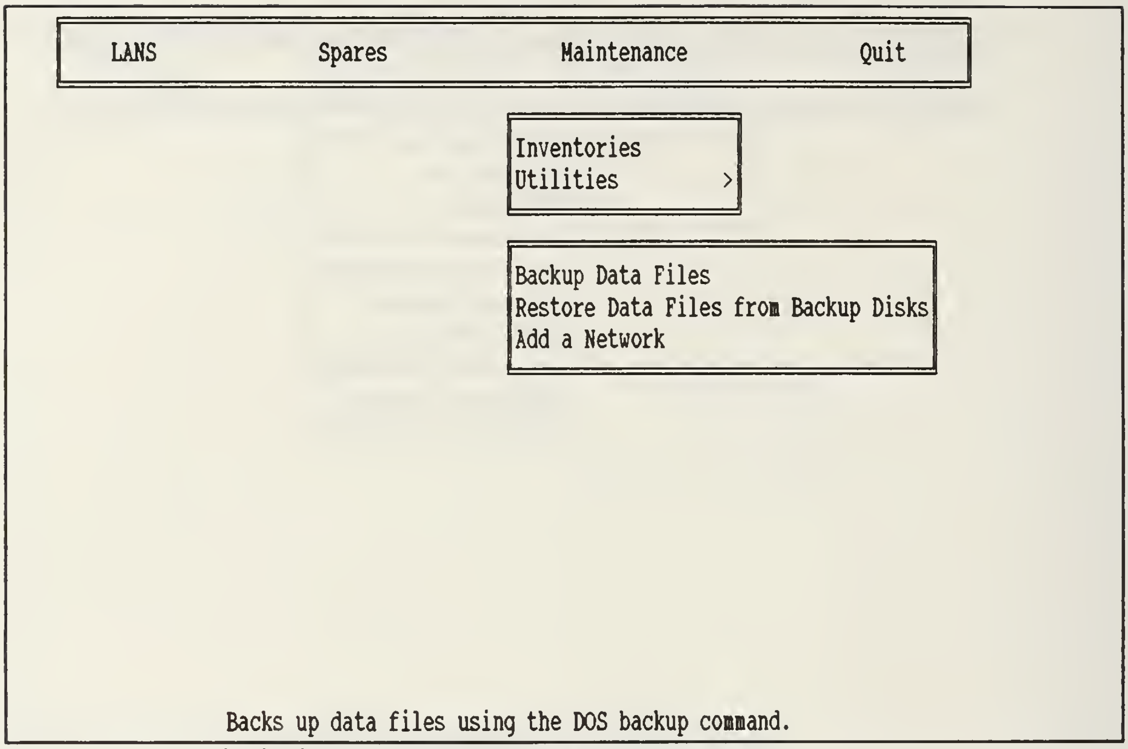
Menu 7 - Maintenance Selection Pull-down

This bar and pull-down combination allows the user to select inventory actions, or several needed utilities. "Inventory" actions, like Spares actions (Menu 5), if a LAN has been selected in Menu 1, are constrained to that LAN until the LAN is unselected. "Inventories" calls Menu 8 and "Utilities" calls Menu 9.

```
LANS          Spares          Maintenance          Quit

                         Inventories       >
                         Utilities

                        Network Inventory
                        Node Inventory
                        Spare Parts Inventory
                        Software Inventory

               Get an inventory on a selected network.
```

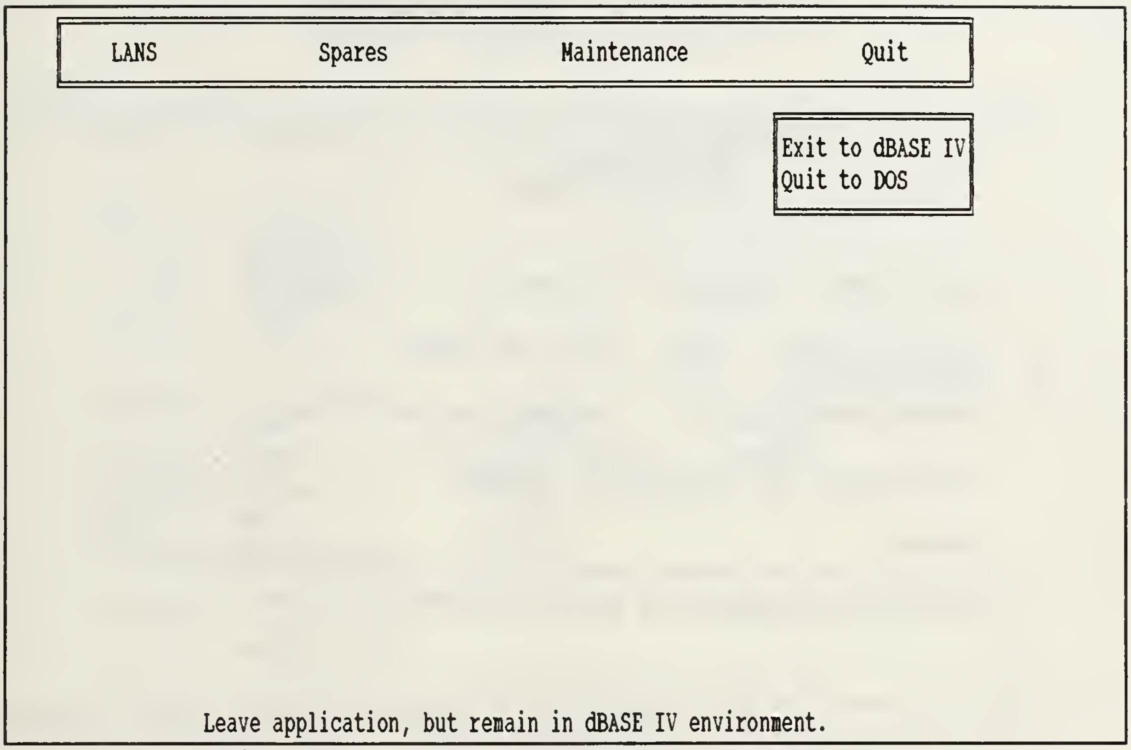
Menu 8 - Inventories Pop-up Submenu

Each of these selections calls an inventory function,and users are granted the opportunity to have the output go to the printer or screen.



Menu 9 - Utilities Pop-up Submenu

The maintenance functions listed each calls sub-routines to "Backup Data Files", "Restore Data Files from Backup Disks", or "Add a Network" to the maintenance system.



Menu 10 - Quit Pull-down Submenu

"Quit" is another combination bar and pull-down. It allows the user to quit the program, but still stay in the dBASE IV environment, or to completely quit the DBMS and return to the machine's operating system.

APPENDIX E - DATA MATERIALIZATIONS

IBM Token-Ring - I227
Node #01U - "NORM12"

Model: 80386 - 33 MHz with 0 K Cache Serial #3333342344

Motherboard Memory: 4096 K BIOS: AMI version 3.10
Expanded Memory: 0 K
Extended Memory: 0 K Keyboard: 101 Key AT Compatible

Video Display: VGA Video Serial # 3234234234

Comments:

The user has the opportunity to leave a word processed note here!

Node Information Form

IBM Token-Ring - I227
Node #01U - "NORM12"

Model: 80386 - 33 MHz with 0 K Cache Serial #3333342344

Drives:

1.2 MB	Floppy Drive	A
1.44 MB	Floppy Drive	B
20 MB	Hard Drive	C
1024 KB	Logical	D

Accessories:

#	Item	Location	Settings
01 2400 bps	Modem	Slot 3 (COM 1)	10110000
02 Dot Matrix	Printer	LPT 1	
03 Serial	Mouse	Com 2	
04 Token-ring	Network Board	Slot 1	

Cable Plant:

Adapter Cable

Sample Node Accessory Display Form

IBM Token-Ring - I227
Node #01U - "NORM12"

Model: 80386 - 33 MHz with 0 K Cache   Serial #3333342344

Accessory Name: Modem   Accessory # 01

Accessory Type: 2400 bps

I/O Port: COM 1

Switch Settings: 1 0 1 1 0 0 X X

Comment:

Sample Accessory Input/Edit Form

LIST OF REFERENCES

Bates, William and Fortino, Andres G., dBASE III PLUS and Local Area Networks, Torrance, CA., Ashton-Tate, 1986.

Date, C. J., An Introduction to Database Systems, Fourth Edition, Reading, MA, Addison-Wesley, 1986.

Foley, George F. IV, Dynamics of FoxBASE+ Programming, Homewood, IL., Dow Jones-Irwin, 1988.

Kroenke, David M. and Dolan, Kathleen A., Database Processing: Fundamentals, Design, Implementation, Science Research Associates, Inc., 1988.

Schneidewind, Norman F., Interview, February 1991.

Simpson, Alan, dBASE III PLUS Programmer's Reference Guide, Alameda, CA., Sybex Inc., 1987.

Simpson, Alan, Understanding dBASE IV, Alameda, CA., Sybex., 1989.

Suriano, Douglas A., "The Design of a Local Area Network Configuration Management System for the Naval Postgraduate School Administrative Sciences Department," Masters Thesis, Naval Postgraduate School, 1989.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Professor Norman F. Schneidewind Administrative Sciences Department Code As/Ss Naval Postgraduate School Monterey, CA 93943-5000	1
4.	Professor Magdi Kamel Administrative Sciences Department Code As/Ka Naval Postgraduate School Monterey, CA 93943-5000	1
5.	Capt Mack L. Brewer 8 Rodeo Court Fredericksburg, VA 22401	1
6.	Commandant of the Marine Corps Code TE 06 Headquarters, U.S. Marine Corps Washington, DC 20380-0001	1